Bag of Tokens (View)

You have an initial **power** of `power`, an initial **score** of `0`, and a bag of `tokens` where `tokens[i]` is the value of the `i`th token (0-indexed).

Your goal is to maximize your total **score** by potentially playing each token in one of two ways:

- If your current **power** is at least `tokens[i]`, you may play the `i`th token face up, losing `tokens[i]` **power** and gaining `1` **score**.
- If your current **score** is at least `1`, you may play the `i`th token face down, gaining `tokens[i]` **power** and losing `1` **score**.

Each token may be played **at most** once and **in any order**. You do **not** have to play all the tokens.

Return *the largest possible **score** you can achieve after playing any number of tokens*.

Example 1:

```
Input: tokens = [100], power = 50

Output: 0

Explanation: Playing the only token in the bag is impossible because you either
have too little power or too little score.
```

Example 2:

```
Input: tokens = [100,200], power = 150

Output: 1

Explanation: Play the 0th token (100) face up, your power becomes 50 and score
becomes 1.

There is no need to play the 1st token since you cannot play it face up to add to
your score.
```

Example 3:

```
Input: tokens = [100,200,300,400], power = 200

Output: 2

Explanation: Play the tokens in this order to get a score of 2:

1. Play the 0th token (100) face up, your power becomes 100 and score becomes 1.
```

```
2. Play the 3rd token (400) face down, your power becomes 500 and score becomes 0.

3. Play the 1st token (200) face up, your power becomes 300 and score becomes 1.

4. Play the 2nd token (300) face up, your power becomes 0 and score becomes 2.
```

Constraints:

- `0 <= tokens.length <= 1000`
- `0 <= tokens[i], power < 10^4`